

線形計画法を用いた電源ノイズ解析高速化手法の検討

胡摩窪 武

鈴木 五郎

北九州市立大学情報メディア工学科 〒808-0135 北九州市若松区ひびきの 1-1

E-mail: e5303701@hibikino.ne.jp, suzuki-g@env.kitakyu-u.ac.jp

あらまし 近年線形計画法を使用して電源ネットに関する電源ノイズ解析を行う手法が提案されている。提案されている既存手法を用いて、マイクロプロセッサの電源ノイズ解析を行おうとした場合、次の2つの問題点があった。1)実行マシン cycle 毎に制約式をたてなければならない、2)ノード毎に最適化しなければならない。そこで結果として膨大な計算量が必要となってしまう。我々は、一回のマシン cycle だけの解析で、しかも目的関数を工夫することで、一度の最適化で所望の解析が可能な手法を提案する。縦・横 20 分割された電源配線 RC モデルを用いて提案手法を評価した結果、既存手法に対し、解析誤差は最大 13%、平均 10%、解析 CPU 時間は 1.1×10^5 分の 1 となった。また SPICE®を用いた解析と比較した結果、解析誤差は最大 20%、平均 16%、および解析 CPU 時間は 83 分の 1 とすることができた。

キーワード 線形計画法、電源ノイズ、マイクロプロセッサ

Power Noise Analysis Acceleration Technique by Linear Programming Method

Takeshi GOMAKUBO and Goro SUZUKI

Department of Information Science, University of Kitakyushu, 1-1 Hibikino, Wakamatsu-ku, Kitakyushu-shi,
Fukuoka-ken, 808-0135 JAPAN

E-mail: e5303701@hibikino.ne.jp, suzuki-g@env.kitakyu-u.ac.jp

Abstract Power noise analysis method using linear programming has been proposed. This method becomes very time consuming in the case of noise analysis coping with application program execution. This is because linear programming has to be executed many times corresponding to the application program execution cycle. We propose novel method to avoid this iteration. Comparing with conventional method, speed can be accelerated by 1.1×10^5 times with small error.

Keyword Linear Programming, Power noise, microprocessor

1. まえがき

設計フローのうち、上流設計での電源ノイズ解析が必要とされている。特に、まだ論理設計が収束していない状態であるが、経験的な機能 block 設計情報をもとにした floor plan が可能な状態での電源ノイズ解析が注目されている[1]。電源設計の変更はレイアウトの手直し、場合によっては論理の修正までも強いるためである。一方電源ノイズ解析には膨大な処理時間を必要としている。電源ネットの等価回路規模が膨大になることが第一の理由であり、第二は Application Program (以下 APP と略す) の処理実行 cycle 毎に電源ノイズがダイナミックに変化することから、実行 cycle 毎の解析が必要となることである。そこで高速・高精度な解析手法として回路縮約技術を使った方法[2][3]、伝送行列を使った方法[4]、あるいは線形計画法(以下 LP と略す)を用いた方法[5][6]などが提案されている。今回は LP を用いた方法をとりあげ、提案

されている既存手法より高速化する手法を提案し、その評価を行う。

2. モデリング

電源ノイズ解析をするにあたって次のように回路のモデル化を行った。例を図 1 に示す。電源ネットの下にはゲートが繋がっており、ゲートが active になるとき貫通電流が流れる。Booth 乗算回路などの単位でゲートに流れる貫通電流の和を等価的にひとつの電流源で表す。

APP が実行されると実行 cycle 毎に active なゲート、inactive なゲートがダイナミックに変化するので、電源ノイズ解析をする場合には cycle 毎に解析が必要である。

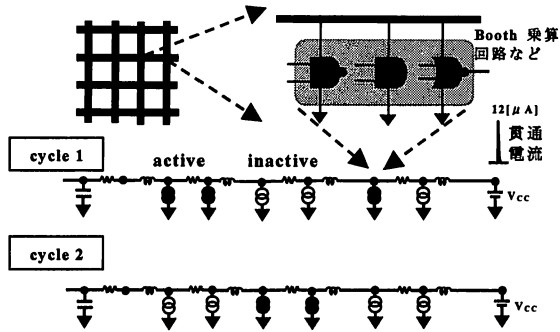


図1 電源ネットと論理ゲートのモデル化

Fig.1 Power net and logic gates modeling

3. 仮想マシンの仕様

仮想マシンと仮想 APP を考えてみる。仮想マシンの電源ネットを図2, 仮想 APP 実行結果を表1に示す。

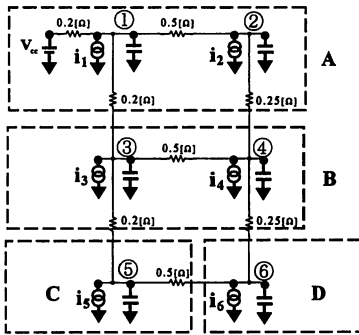


図2 仮想マシン電源ネット

Fig.2 Power net of virtual machine

表1 仮想 APP 実行結果

Table 1 Virtual APP execution result

cycle block	1	2	3	4	5	6	7	8	9	10	...
A	●		●	●				●			...
B		●		●		●			●	●	...
C	●		●	●		●					...
D				●	●		●	●		●	...

●active 空白は inactive

縦2×横3の電源ネットの下に A, B, C, D の floor plan 上の block が存在し、各ノードにはノイズ源となる電流源が繋がっている。図2の①はノード番号である。

電流源の電流値の最大は初めに経験値より決める。ここでは 12[μA]とした(図3の左)。ところが、表1

より、cycle4 に全ての block が同時に active になり許容最大電力をオーバーするため、ノード②、ノード⑤に関しては 10[μA]に変更した(図3の右)。

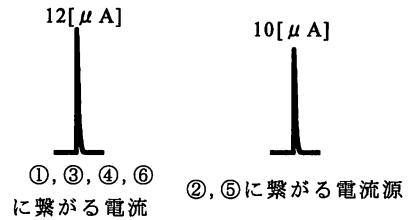


図3 電流源波形1

Fig.3 Current source wave form 1

仮想 APP 実行結果より、cycle 毎に見て同時に active になる block は cycle 1, 2, 4, 5, 6, 8, 10 の 7case ある。この7つの cycle を case1~7とする。

仮想マシンを SPICE®で解析するときには case 毎にどのノードに電流源を繋ぐかを変える必要があるので 7回の回路シミュレーションが必要となる。

4. 既存手法

LP を用いた解析手法が提案されている[5]。その手法はノード毎に LP を適用し、それを全ての case で行うというものである。

例えば、case1 では block A の中にある2つの電流源と block C の中にある1つの電流源が active となる。ノード①に繋がる電流源の電流値を i_1 とすると、ノード①に繋がる電流源 i_1 から流れ出る負の電流は左右方向および下方向に流れ出る。よってキルヒホフの電流則より i_1 は式(1)で表すことができる (v_i は V_{cc} からの電圧降下分)。

$$i_1 = \frac{V_{cc} - (V_{cc} - v_1)}{0.2} + \frac{(V_{cc} - v_2) - (V_{cc} - v_1)}{0.5} + \frac{(V_{cc} - v_3) - (V_{cc} - v_1)}{0.2} \quad (1)$$

また、 i_1 の最大値を I_1 とすると式(2)となるので、ノード①は式(1)、(2)より式(3)と表すことができる。

$$i_1 \leq I_1 \quad (2)$$

$$12v_1 - 2v_2 - 5v_3 \leq I_1 \quad (3)$$

active な電流源が繋がっているノード②、⑤も同様に表すことができる。一方、inactive な電流源が繋がっているノード③、④、⑥では電流源から流れ出る電流が0なのでイコール0で表現できる。これらをまとめたものが式(4)である。

$$\left. \begin{array}{l} \text{ノード①} \quad 12v_1 - 2v_2 - 5v_3 \leq I_1 \\ \text{ノード②} \quad -2v_1 + 6v_2 - 4v_4 \leq I_2 \\ \text{ノード⑤} \quad -5v_3 + 7v_5 - 2v_6 \leq I_5 \\ \text{ノード③} \quad -5v_1 + 12v_3 - 2v_4 - 5v_5 = 0 \\ \text{ノード④} \quad -4v_2 - 2v_3 + 10v_4 - 4v_6 = 0 \\ \text{ノード⑥} \quad -4v_4 - 2v_5 + 6v_6 = 0 \end{array} \right\} (4)$$

式(4)の制約を基に各ノード電圧降下最大値を個別に求めるので、ノード数 $6 \times \text{case 数 } 7$ の 42 回 LP を実行する必要がある。

この手法は LP の実行回数が多いという問題点がある。

5. 提案手法

5.1. 提案手法 1

そこで、ノード毎に LP を適用し、それを 1 回で済ませる手法を考えた。そのために 2 つの手段を考えた。

手段の 1 つ目は排他的に active になる block の group 化である。表 2 で block A, B に注目すると case 毎に見たとき排他的に active になっていることが分かり、同様に block C, D も排他的に active になっていることが分かる。従って、block A, B を G1, block C, D を G2 と group 化し group 毎で制約式を求めることにした。

表 2 Block の group 化 1

Table 2 Block grouping 1

block	case							group 名	block 最大電流	group 最大電流
	1	2	3	4	5	6	7			
A	●		●			●		G1	$I_1 + I_2 = 22$	$\max\{22, 24\} = 24$
B		●			●		●		$I_3 + I_4 = 24$	
C	●					●		G2	$I_5 = 10$	$\max\{10, 12\} = 12$
D				●		●	●		$I_6 = 12$	

G1 に関して見ると block A の最大電流はノード①, ②に繋がる電流源が active なときの $22[\mu A]$, block B では同様に $24[\mu A]$ であり、block A, B は排他的に active になるので、G1 の最大電流は $24[\mu A]$ となる。よって、G1 に含まれるノード①, ②, ③, ④が active なときの制約である式(5)の左辺の合計が 24 以下であると考え、式(6)の制約を考えた。同様に、G2 の最大電流は $12[\mu A]$ となるので、式(7)の左辺の合計が 12 以下と考え、式(8)の制約を考えた。この制約を制約 1 とする。

$$\left. \begin{array}{l} 12v_1 - 2v_2 - 5v_3 \leq I_1 \\ -2v_1 + 6v_2 - 4v_4 \leq I_2 \\ -5v_3 + 7v_5 - 2v_6 \leq I_5 \\ -5v_1 + 12v_3 - 2v_4 - 5v_5 \leq I_3 \\ -4v_2 - 2v_3 + 10v_4 - 4v_6 \leq I_4 \end{array} \right\} (5)$$

$$5v_1 + 5v_3 + 4v_4 - 5v_5 - 4v_6 \leq 24 \quad (6)$$

$$\left. \begin{array}{l} -5v_3 + 7v_5 - 2v_6 \leq I_5 \\ -4v_4 - 2v_5 + 6v_6 \leq I_6 \end{array} \right\} (7)$$

$$-5v_3 - 4v_4 + 5v_5 + 4v_6 \leq 12 \quad (8)$$

手段の 2 つ目は実行 cycle における active, inactive をまとめて不等式で表現することである。APP を実行したとき実行 cycle 毎に電流源が active か inactive かがダイナミックに変わる (図 4)。

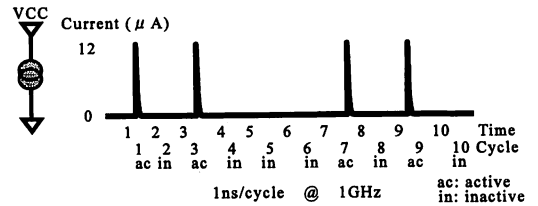


図 4 電流源波形 2

Fig.4 Current source wave form 2

電流源が active なときは式(9)のように不等式で表すことができ、inactive なときは式(10)のようにイコール 0 と表すことができたが、active, inactive を合わせて表現すると式(9)のように不等式で表すことができる。

$$12v_1 - 2v_2 - 5v_3 \leq I_1 \quad (9)$$

$$12v_1 - 2v_2 - 5v_3 = 0 \quad (10)$$

全てのノードにて同様のことをすると式(11)の制約を考えることができる。この制約を制約 2 とする。

$$\left. \begin{array}{l} 12v_1 - 2v_2 - 5v_3 \leq I_1 \\ -2v_1 + 6v_2 - 4v_4 \leq I_2 \\ -5v_3 + 7v_5 - 2v_6 \leq I_5 \\ -5v_1 + 12v_3 - 2v_4 - 5v_5 \leq I_3 \\ -4v_2 - 2v_3 + 10v_4 - 4v_6 \leq I_4 \\ -5v_3 + 7v_5 - 2v_6 \leq I_5 \\ -4v_4 - 2v_5 + 6v_6 \leq I_6 \end{array} \right\} (11)$$

制約 1 と制約 2 を基に各ノードの電圧降下最大値を

個別に求めることにした。よって、ノード数 6×1 の 6 回 LP を実行する。

5.2. 提案手法 2

更に LP 実行回数を減らすために、全ノードまとめて LP を適用し、それを 1 回で済ませる手法を考えた。そのために 2 つの手段を考えた。

手段の 1 つ目は全ノードの電圧値の和を目的関数にすることである。ここで 2 変数の問題を考えてみる。図 5 の塗りこまれた領域を LP 制約領域とする。軸上にない頂点でのなす角が鈍角である場合 type1 (図 5 の左) は、電圧値を個別に目的関数としたときと電圧値の和を目的関数としたときで最大値に誤差が生じる。一方、軸上にない頂点でのなす角が鋭角である場合 type2 (図 5 の右) は、電圧値を個別に目的関数としたときと電圧値の和を目的関数としたときで最大値は一致する。

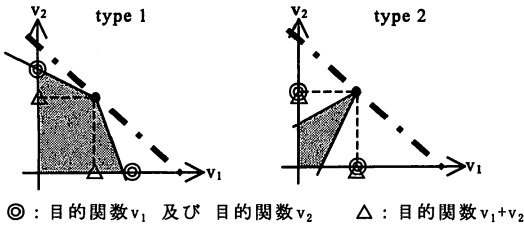


図 5 2 変数の制約領域

Fig.5 Two dimensional constraint region

ここで、制約 2 は type2 を満足することを説明する。図 6 の回路のときノードは 3 つあるので制約 2 は式(12)と表せる。抵抗値は全て $1[\Omega]$ 、電流源の電流値の最大を全て $1[\mu A]$ とする。

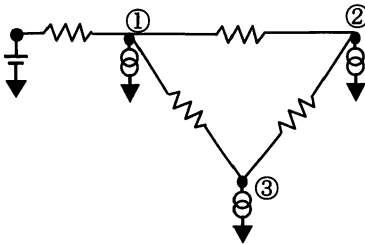


図 6 例題回路

Fig.6 Example circuit

$$\left. \begin{array}{l} \text{ノード①} \quad 3v_1 - v_2 - v_3 \leq 1 \quad \text{平面1} \\ \text{ノード②} \quad -v_1 + 2v_2 - v_3 \leq 1 \quad \text{平面2} \\ \text{ノード③} \quad -v_1 - v_2 + 2v_3 \leq 1 \quad \text{平面3} \end{array} \right\} (12)$$

3 変数なので 3 次元で考えると各制約は面をもつ。ここで、平面 1 と平面 2 に注目し図 7 の塗りこまれた領域が制約領域であるとき、平面 1 と平面 2 がなす角

θ_{12} を求めるために、まず法線ベクトルのなす角 α_{12} を計算する。各法線ベクトルは式(13)で表されるので $\cos \alpha_{12}$ は式(14)の 1 行目のように負の値をとるので、

$$\frac{\pi}{2} < \alpha_{12} \leq \pi \text{ となる。} \quad \alpha_{12} \text{ は鈍角であり、}$$

$\theta_{12} + \alpha_{12} = \pi$ であるので θ_{12} は鋭角であることが分かる。平面が 3 つなので平面同士がなす角は 3 つあるが、同様に全て鋭角であることが分かる。つまり制約 2 は type2 を満たす。

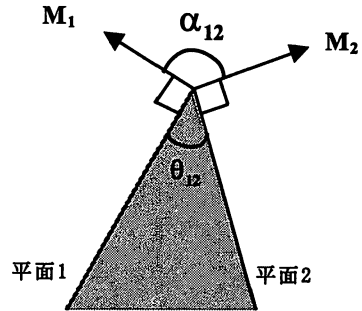


図 7 平面 1 と平面 2 の作る制約領域

Fig.7 Constraint region by two planes

$$\left. \begin{array}{l} \mathbf{M}_1 = (3, -1, -1) \\ \mathbf{M}_2 = (-1, 2, -1) \\ \mathbf{M}_3 = (-1, -1, 2) \end{array} \right\} (13)$$

$$\left. \begin{array}{l} \cos \alpha_{12} = \frac{\mathbf{M}_1 \mathbf{M}_2}{|\mathbf{M}_1| |\mathbf{M}_2|} = -0.49 < 0 \\ \cos \alpha_{23} = \frac{\mathbf{M}_2 \mathbf{M}_3}{|\mathbf{M}_2| |\mathbf{M}_3|} = -0.5 < 0 \\ \cos \alpha_{31} = \frac{\mathbf{M}_3 \mathbf{M}_1}{|\mathbf{M}_3| |\mathbf{M}_1|} = -0.49 < 0 \end{array} \right\} (14)$$

しかし、制約 1 は type2 を満たさない。そこで、2

がある。電源ネットは縦 20×横 20 であり、命令は 8 種類あり、CPI=8 である。これに畳み込み演算を行う 40 行の APP を実行させる。実行 cycle 総数は 8cycle×40 行の 320cycle であるが、cycle 毎に見て active になる block は表 3 の 19case であった。表 3 の R0~R5 に注目すると case 毎に見たとき排他的に active になっているので R0~R5 を G1 とした。同様に INST と ECC を G2、ALU を G3、SEL.CC.PC を G4 とした。

踏襲しており，提案手法 3 は提案手法 1, 2 を踏襲しているため，提案手法 1, 2, 3 の順に CPU 時間が改善されていることが分かる．誤差は対 SPICE 比を求めるために式(21)とし，提案手法 3 では平均誤差は 16.3[%]となった．

grid verification, " IEEE/ACM International on Computer-aided design, pp358-364, San Jose, CA, Nov. 2005.

[7] 鈴木他, “ECC 回路内蔵 16 ビットマイコンの設計,” 信学総大, C-12-7, 2006.

表 4 評価結果

Table 4 Evaluation result

	CPU 時間[s]	平均誤差[%]	最大誤差[%]
SPICE®	18.35	—	—
既存手法	24,409	5.6	6.2
提案手法 1	2,480	14.9	28.8
提案手法 2	5.23	16.3	19.9
提案手法 3	0.22	16.3	19.9

$$\text{誤差}[\%] = \frac{|\text{SPICE 値} - \text{解析値}|}{\text{SPICE 値}} \times 100 \quad (21)$$

7. むすび

LP を用いた電源ノイズ解析方法を取りあげ，提案されている既存手法より高速化する手法を提案し，その評価を行った．提案手法 1 及び提案手法 2 を踏襲した提案手法 3 は，SPICE®の 80 倍，既存手法の 1.1×10⁵ 倍高速化することができた．精度は，対 SPICE®比で平均 16.3%であった．

謝辞 本研究は㈱半導体理工学研究センター STARC の支援の下で行った．客員研究員等関係各位に深く感謝する．

文 献

[1] F. N. Najm, “Vectorless Power Grid Verification, ” ACM/IEEE Design Automation Conference, Tutorial 4, June 2007.

[2] 鈴木他, “MOR 法の電源ノイズ解析システムへの適用,” 電子情報通信学会論文誌 A, Vol.J91-A, No.6, pp.695-698, June 2008.

[3] G. Suzuki, “VLSI Power Wiring Noise Analysis Using MOR Method,” Proc. of IEEE 7th ASICON, pp.1186-1189, Oct. 2007.

[4] Y. Murasaka, M. Nagata, T. Ohmoto, and A. Iwata, “Chip-Level Substrate Noise Analysis with Network Reduction by Fundamental Matrix Computation, ” Proc. of International Symposium on Quality Electronic Design, 4C.3, pp.482-487, March 2001.

[5] D. Kouroussis and F. N. Najm, “A static pattern-independent technique for power grid voltage integrity verification, ” ACM/IEEE Design Automation Conference, pp.99-104, June 2003.

[6] D. Kouroussis, I. A. Ferxli and F. N. Najm, “Incremental partitioning-based vectorless power